CLAIMS

What is claimed is:

1. A method of routing traffic between a source router and a destination router within a multi-path network, comprising:

determining multiple loop-free paths of unequal cost to a destination router in response to long-term link-cost information;

allocating a route to said destination router along one of said multiple loop-free paths; and

adjusting routing parameters available at each router in response to short-term link-cost information to incrementally adjust route allocation.

2. A method as recited in claim 1, wherein said long-term link-cost information is determined within said routers by executing heuristic programming to update successor set information at each router.

3. A method as recited in claim 1, wherein said short-term link-cost information is determined within said routers by executing heuristic programming to update routing parameters at each router.

4. A method as recited in claim 1, wherein said short-term link-cost information is computed by each router in response to information received within link-state update messages, or equivalent.

5. A method as recited in claim 4, wherein said link-state update message indicates that an addition, deletion, or change in link-costs has occurred.

6. A method as recited in claim 1,

wherein allocating of said route does not require global synchronization on the network;

whereby said routing method is able to respond to rapidly-changing traffic conditions within said network.

7. A method as recited in claim 1,

wherein said short-term link-cost information is gathered at intervals of length T_S; and

wherein said short-term link-cost information is utilized to adjust the routing-parameters of routers along said loop-free path.

8. A method as recited in claim 1,

wherein said long-term link-cost information is gathered at intervals of length T_l;

wherein said long-term link-cost information is used to update successor set information for each router; and

wherein said long-term link-cost information is utilized for initializing a near-optimum routing path.

9. A method as recited in claim 1, wherein short-term and long-term link-cost information is maintained in tables at each router.

10. A method as recited in claim 9, wherein said tables comprise:

a main topology table T^i, or equivalent, in which information is maintained about the characteristics of each link known to router i;

a neighbor topology table T_k^i, or equivalent, in which information is maintained about each neighbor k;

a distance table in which distance information is maintained from router i to each destination based on the topology in said main topology table;

a routing table in which information about routing paths to the destinations are maintained; and

a link table in which link-cost information l_k^i is maintained for each neighbor k.

11. A method as recited in claim 10, wherein the routing path information maintained in said routing table comprises:

successor set S_j^i to each destination j; and

feasible distance FD_j^i.

12. A method as recited in claim 1,

wherein the traffic allocation on a link substantially satisfies the following equation: $\phi^i_{jk} = \psi\left(k, \left\{D^p_j + l^i_p \mid p \in N^i\right\}, \left\{\phi^i_{jp} \mid p \in N^i\right\}\right) \quad k \in N^i$, where O^i_{jk} is the routing path parameter and where ψ is a flow allocation function.

13. A method as recited in claim 1, wherein determining of multiple loop-free paths is performed according to an approximation of minimum delay routing.

14. A method of approximating minimum delay routing between a source and a destination within a computer network having a plurality of available paths, comprising:

deriving an approximation to the Gallager minimum-delay routing problem to determine near-optimal routes between said source and said destination; and

allocating routes according to said approximation based on link-state information so as to provide multiple paths of unequal cost to each destination that are loop-free.

15. A method as recited in claim 14, wherein said link-state information comprises:

long-term link information containing information about the near-shortest routing path;

said long-term link information further containing information about successor

sets at each router; and

short-term link information containing recent information about that state of the links for use in adjusting routing parameters at each router.

16. A method as recited in claim 15, wherein said short-term link information is updated more frequently than the long-term link information.

17. A method as recited in claim 15, wherein said short-term link-cost information is computed by each router in response to information received within link-state update messages, or equivalent.

18. A method as recited in claim 15, wherein said link-state update messages indicate that an addition, deletion, or change in link-costs has occurred.

19. A method as recited in claim 14,

wherein the derivation of said near-optimal routes does not require global synchronization on the network;

whereby said routing method can respond to rapidly-changing traffic conditions.

20. A method as recited in claim 19,

wherein global variables for the network do not need to be maintained.

21. A method as recited in claim 15, wherein short-term and long-term link-cost information are maintained in a series of tables at each router.

22. A method as recited in claim 21, wherein said tables, comprise:

a main topology table T^i, or equivalent, in which information is maintained about the characteristics of each link known to router i;

a neighbor topology table T_k^i, or equivalent, in which information is maintained about each neighbor k;

a distance table in which distance information is maintained from router i to each destination based on the topology in said main topology table;

a routing table in which information about routing paths to the destinations are maintained; and

a link table in which link-cost information l_k^i is maintained for each neighbor k.

23. A method as recited in claim 22, wherein the routing path information maintained in said routing table comprises:

successor set S_j^i to each destination j; and

feasible distance FD_j^i.

24. A method as recited in claim 23, wherein said tables are maintained within by the executing of procedures within said routers, comprising:

a main topology update procedure (MTU), or equivalent;

a multiple-path partial-topology dissemination procedure (MPDA), or equivalent, which is invoked when an event occurs to disseminate topology information to routers;

an initializing procedure for said multiple-path partial-topology dissemination procedure (INIT-PDA), or equivalent;

a neighbor topology update procedure (NTU) for updating the topology of neighboring routers;

initial route assignment procedure (IH) for allocating a near-optimal initial route between a source and a destination according to said long-term link-cost information; and

an incremental loading procedure (AH) which adjusts routing parameters according to said short-term link-cost information.

25. A method of allocating loop-free multi-path traffic routing between routers within a network having a plurality of routing paths between said source and said destination, comprising:

computing multiple loop-free paths between said routers;

distributing traffic over said loop-free paths; and

updating link costs associated with said paths to optimize local traffic flow.

26. A method as recited in 25, wherein the computing of said loop-free paths comprises:

computing D_j^i using a shortest-path algorithm, or equivalent, based on link-state information; and

computing S_j^i by extending said shortest-path algorithm to support multiple successors along the loop-free path to each destination.

27. A method as recited in 25, wherein distributing traffic over said paths comprises:

executing a heuristic algorithm *IH*, or equivalent, to determine an initial load assignment; and

periodically executing a heuristic algorithm *AH*, or equivalent, to adjust the incremental load.

28. A method as recited in 25, wherein updating link costs associated with said paths to optimize local traffic flow comprises:

estimating marginal delay along each path; and

communicating link-state update messages (LSUs) which contain information about said marginal delay along said paths.

29. A method of approximating minimum delay between routers within a computer network having a plurality of available paths by executing a distributed routing algorithm, comprising:

determining a set of marginal distances $D_j^i = min\left\{D_j^k + l_k^i \mid k \in N^i\right\}$;

finding a feasible distance FD_j^i which satisfies the relationship $FD_j^i \leq D_{ji}^k$ wherein $k \in N^i$;

determining a successor set $S_j^i = \left\{k \mid D_{jk}^i < FD_j^i \wedge k \in N^i\right\}$ or equivalent; and

allocating traffic $\phi_{jk}^i = \psi\left(k, \left\{D_j^p + l_p^i \mid p \in N^i\right\}, \left\{\phi_{jp}^i \mid p \in N^i\right\}\right)$ wherein $k \in N^i$, or equivalent along said routes;

30. A method of assuring loop-free routing by a router executing a given routing algorithm and operated within a network having multiple paths between sources and destinations, comprising:

finding a feasible distance FD_j^i which satisfies the relationship $FD_j^i \leq D_{ji}^k$ wherein $k \in N^i$;

determining a successor set $S_j^i = \left\{k \mid D_{jk}^i < FD_j^i \wedge k \in N^i\right\}$ or equivalent; and

wherein any routing path satisfying the above equations is assured of being a loop-free routing path within said network.